Context-agnostic writing in modular documents

Tony Self

Tony Self has worked as a technical communicator for 30 years, with the last 20 of those years specifically in the areas of help systems and electronic documents. In 1993, Tony founded HyperWrite, a hypertext and technical documentation company based in Melbourne. In addition to consultancy work, Tony lectures in the Technical Communication programme at Swinburne University of Technology, and holds a Graduate Certificate in Teaching and Learning and a Graduate Diploma in Technical Communication. He is a Fellow of the Institute of Scientific and Technical Communicators, member of the OASIS DITA Technical Committee, and chair of the DITA Help Subcommittee.

Abstract

In the field of technical communication, there is a migration (driven by the need for efficiency) from style-based, document-centric writing approaches to topic-oriented, modular authoring writing techniques (Houlihan, 2008, p. 4). One of the challenges of working within a modular, structured, semantic authoring environment is the imperative to remove context from the writing to enable re-use. In a modular writing environment, a single source or repository of topical information modules are assembled into different publications for delivery in different reading formats. The ability to re-use the same content modules is especially important for organisations managing large documentation suites, such as motor vehicle manufacturers producing documents with sizable proportions of common content.

This paper argues that by adopting context-agnostic writing techniques for topic-based modular documentation, technical writers can improve content re-use and achieve greater efficiency through the technical documentation life cycle without significantly compromising quality. Context-agnostic writing techniques focus on the separation of context from content. This paper will be of interest to people working in the field of technical communication, and also to people engaged in other areas of written communication practice.

Introduction

Technical documentation is a broad term which encompasses computer software and hardware user guides, corporate policy and procedure manuals, Help systems, scientific and medical publications, engineering manuals, machinery instructions, reference materials, and many other forms of non-fiction, corporate, product and business discourse. The word “technical” in the term is sometimes misleading, because the term also encompasses some forms of organisational communication. Technical documentation is commonly produced by professional technical writers. For many years, technical writers have followed a document-centric, linear, narrative writing paradigm, treating a manual as a self-contained and isolated work (Rockley, 2001). Before computerisation, technical writers wrote drafts in longhand before sending them for typing, rewriting, editing, reviewing and typesetting. When word processing software tools were adopted by technical writers, document-centric authoring programs such as WordPerfect, Word and FrameMaker allowed the same style-based, document-centric paradigm to be used, but with the technical writer taking over the

former roles of typists, typesetters, layout artists and, in some cases, printers (O'Hara, 2001).

When topic-based, modular writing techniques became known in the early 1990s, the concept of *single sourcing* became practicable (Robidoux, 2008, p. 111). Single sourcing is "a method for developing re-usable information" (Ament, 2003, p. xiii), where re-usable document modules are assembled to form publications, with different combinations of modules resulting in separate publications. *Modular writing* is a technique that makes single-sourcing possible. Modular, or *topic-based*, writing, is a style of document design and architecture where content is structured into independent small modules (*topics*) which can be assembled into one or many larger texts, such as a books, Web sites and Help systems. The advent of documentation technologies based on eXtensible Markup Language (XML), at the start of 21st century, expanded the possibilities for single-sourcing and re-use, and led to the adoption of the philosophy of *separation of content and form* through semantic mark-up (Sapienza, 2004, p. 400). The *Darwin Information Typing Architecture (DITA)* is an XML-based documentation methodology developed explicitly for technical documentation. DITA is primarily a semantic authoring mark-up language, incorporating the ideas of topic-based, modular architecture, semantic mark-up, and XML-based standard information structures. (XML is a set of standards for the categorisation, storage and retrieval of all forms of structured information.)

This paper argues that, by adopting context-agnostic writing techniques for topic-based modular documentation, technical writers can improve content re-use and achieve greater efficiency through the technical documentation life cycle without significantly compromising quality. Context-agnostic writing techniques focus on the separation of context from content. This separation is a further extension of the philosophy of the separation of content and form. By minimising context at the topic level, and abstracting it to the document or publication level, the opportunities for topic re-use are maximised.

Modular writing

Topic-oriented, modular writing is an approach or a technique, rather than a technology in its own right. The modular writing methodology is simply enabled by technologies such as XML. The change in writing technique from document-centric to modular can be a challenge to many technical writers. A 2009 survey of technical writers using DITA observed that the biggest challenge for writers is caused by the topic-oriented paradigm, and that DITA necessitated "making a major change in the document-centric mindset" (WritePoint, 2010, p. 12).

It is not just the topic-oriented paradigm that presents a challenge to writers. Being able to re-use, re-purpose and re-assemble topics into different publications in different forms requires many other changes in writing (Williams, 2003). The separation of content and form is alien to those writers accustomed to working in format-driven environments such as Microsoft Word, where *WYSIWYG (What You See Is What You Get)* is a fundamental principle. Rich authoring environments in semantic mark-up languages use an alternative *WYSIOO* approach—What You See Is (just) One Option (O'Keefe, 2006).

Separation of content, form, and context

The use of semantic mark-up in DITA, where text elements are marked up based on their meaning, allows the content to be completely separated from its rendition and display to the reader. For example, a term is marked up as a <term> and a citation as a <cite>, and no information about how those elements will be displayed is stored in the content. Stylistic (display) rules are applied when the DITA content is *transformed* into a reading format, such as HTML or paper. In a DITA workflow, documents are created as collections of modular, re-usable topic files, and mechanisms allow not only the format to be separated from the content, but also the context. The same topic may be a section in the context of one publication, but a sub-section in the context of another. The intermingling of content, format and context in a style-based document workflow essentially eliminates the possibility of re-use. Once a paragraph is styled as having a 13 cm left margin, it cannot be used on paper 12 cm wide. A phrase marked up in italic won't render as italic on a reading device that doesn't support italic. But a citation identified as a citation in a DITA topic can be processed to italic by one transformation process, to bold red by a different transformation process, and to synthesised voice by another transformation process.

One of the more difficult changes to writing technique when moving from linear to modular writing is the removal of as much context as possible from the text. For example, the use of phrases such as "as shown above" and "in the following diagram" will not be valid if the referenced content is not included in all output publications. Well-written topics (and smaller blocks of text) with minimal context can be re-used in many publishing contexts. Writing in such an approach can be referred to as *context-agnostic* or *context-neutral* writing.

The *agnostic* descriptor has been introduced into information technology terminology to make the distinction between systems that are unaware of the platform on which they run, and those that are aware of the platform but adopt a non-preferential, neutral stance with respect to platform (*neutral*). In other words, *agnostic* means no position (unknowable), while *neutral* is a deliberate position. Terms such as *tool-agnostic*, *schema-agnostic*, *implementation-agnostic*, *industry-agnostic*, *technology-agnostic*, *system-agnostic*, and *language-agnostic* illustrate the usage.

In technical and marketing literature, *agnostic* often has a meaning close to "independent"—for example, "platform agnostic" or "hardware agnostic". (Agnosticism, (n.d.))

In this paper, I prefer the term *context-agnostic* because it describes the scenario where the author is unaware of the contexts in which a content module will be used.

Context-agnosticism: Utility, not poetry

Petelin and Durham (1992, pp. 20-22) define three types of context that need to be considered by a writer: "the context of culture, the context of organisations, and the brief context of particular situations". These three types of context need to be separated from content in the writing workflow for the content to be suitable for re-use in multiple contexts. Separating cultural context might mean avoiding culture-specific terms, such as "bonnet" and "hood". Separating organisation context might mean avoiding product names. Separating situational context might mean standardising writing style.

O'Neill (2002) identified the following types of contextual information that needs to be minimised in order to produce truly global, modular information:

- product and company names;
- organisation-specific terminology;
- different ways of writing; and
- inconsistent standards application in authoring tools.

Writing content that is abstracted from its delivery, presentation, structure and context may result in a sacrifice of the look-and-feel of the end product. However, compromise in technical documentation, in Web design, and in most other fields of communication is normal (Hackos, 1994, pp. 214-215). Content-agnostic writing is a typical information architecture decision which does sacrifice some aesthetics for re-usability and efficiency. As put succinctly by Ellen McDaniel in a conference paper on structured authoring in XML, context-neutral authoring is about "utility, not poetry" (McDaniel, 2005, p. 9).

The view that writing without knowledge of context is a valid approach is not universally accepted. Many of the arguments for and against writing for single-source publishing were summarised by Robidoux (2008, pp. 110-111) in a paper discussing the challenges of developing a curriculum for teaching single-sourcing to technical writers.

Although there is evidence that single-sourcing and content re-use reduces writing costs (Houlihan, 2010), there is not yet a large body of evidence to suggest that context-agnostic writing leads to equivalent (or superior) quality of the reading experience itself. However, there are some indications that the benefits of modular writing, such as the ability to customise the modules delivered (online) to suit the needs of an individual reader, do lead to a better reading experience. For example, Hackos and Hedlund (2001, p. 2) point to a case study where, from the perspective of a reader (or information "user"), information can be "delivered in a different way that makes more sense and is more usable in the context of its use".

My experience with single-sourcing has led me to the belief that choosing a context-agnostic writing approach is a sound, pragmatic decision. Context-agnostic writing does not mean the removal of context, but the separation of context from content.

Techniques for reducing context

This paper suggests that in order to write in a context-agnostic way, the following techniques can be used:

- removing or minimising context phrasing (e.g. "Have your vehicle inspected . . .");
- avoiding terms specific to organisations, industries, geography and culture;
- using filtering to selectively remove conditional content from the deliverable document;
- removing branding and other context from graphics;
- removing sequence context and enumeration from topics, and automatically applying sequence in the publishing process; and

- externalising context (by storing context in document maps rather than topics, and using devices such as relationship tables, variables and indirection).

Simplistic removal of context

Some contextual phrasing, such as product and company names, can be removed from text without significant compromise to its readability.

For example, a sentence in the User Guide for (the fictitious) ProductA of "Use ProductA to create a project file to manage all the files in your system" can be re-written as "Use this product to create a project file to manage all the files in your system". The re-written sentence is then able to be re-used in the User Guide for the similar ProductB.

Likewise, a sentence in a car owner's manual of "your Subaru Impreza is fitted with a supplemental restraint system" can be re-written as "your car is fitted with a supplemental restraint system". This change removes the context binding the sentence to Subaru Impreza cars only, so that the sentence may be re-used elsewhere, such as in the owner's manual for a Saab 9_2 (a re-badged Subaru Impreza WRX). However, there is more context that can be removed from the same sentence. The re-written sentence mentions "car", which restricts the use of the sentence to documents concerning cars. Using "vehicle" instead would broaden the potential use of the sentence to documents concerning cars, trucks and boats.

Writers working in a modular writing environment do not need to know where a topic they are writing is intended to be used, or may one day be used. In fact, it may benefit the writer not to know what a topic is intended for, making it easier to write without a context. It is also good practice to be as generic as possible. In the example above, a writer knowing that Subaru doesn't currently make boats or trucks may be tempted to use "car" rather than the more agnostic "vehicle". However, using the narrower "car" context may close off some future re-use opportunities.

In the following example, reference is made to a Subaru dealer; this context limits the application of this paragraph to Subaru manuals (assuming a *variable* has not been used). A context-neutral alternative sentence might read: "Have your vehicle inspected at your local dealer".

> If your vehicle has sustained impact, this may affect the proper function of the Subaru advanced frontal airbag system. Have your vehicle inspected at your SUBARU dealer.

In some cases, the overuse of generic terms may cause misinterpretation of the text. When the context is removed in the example above, a reader may interpret the phrase "your local dealer" as meaning the nearest motor vehicle trader, rather than the intended meaning of the nearest Subaru service centre. Care has to be taken to find the right balance so that the removal of context does not result in the removal of meaning.

DITA provides other mechanisms to allow better management of terminology such as product and company names through features such as variables. This permits document-specific terms to be identified in the topics, and then substituted with a contextually applicable alternative when the document is published to a deliverable format.

Supra-organisation specific terminology

Although not mentioned by O'Neill (2002) in the list of types of contextual information, supra-organisation specific terminology is also a common type of contextual

information that would need to be removed to maximise re-use potential. For example, referring to tax authorities using the US term "IRA" binds the use of the text to a United States context (US Government Internal Revenue Agency).

The following text (from a *Subaru Impreza MY06 Manual*) is largely context-agnostic.

> Your vehicle is equipped with a seatbelt warning device at the driver's seat, as required by current safety standards. There is a seatbelt warning light in the combination meter.

The authors have correctly used "your vehicle", instead of "your Impreza" or "your Subaru". Rather than quoting a specific law or safety regulation, the authors have chosen the neutral phrase "current safety standards". Using more specific (and contextual) language such as "as required by Australian Design Rules" would require the text to be re-written and replaced for every market other than Australia. It might also lead to unnecessary research, such as determining what safety standards are used in Papua New Guinea for the manuals delivered to that country. The reader of the manual, in this case the driver of a Subaru Impreza, just needs to know that the car is fitted with a seatbelt warning device. It could be argued that the phrase "as required by current safety standards" could be removed entirely.

Compare that context-neutral approach with the following paragraph, which can only be understood in the context of a US-based reader:

> Your vehicle is equipped with a Subaru advanced frontal airbag system that complies with the new advanced frontal airbag requirements in the amended Federal Motor Vehicle Safety Standard (FMVSS) No. 208.

To a driver in Malaysia, a reference to a (presumably) US Federal standard is irrelevant at best, and confusing at worst. Further, the use of the word "new" restricts the use of the sentence to a time frame; in a year's time, FMVSS 208 will surely no longer be new!

Use of terms specific to a culture or a geographical audience is another example of context-heavy writing. The phrase "trunk lid" may be understood by North American English-speaking audiences, but is not readily understood by many other English-speaking audiences, who know the component as a "boot lid". Using a variable for the term, using an alternative neutral term (if possible), or including both terms are some methods to dilute the context.

Filtering out context

Conditionalising content helps avoid another obstacle to re-use. To avoid having complicated text blocks listing conditions when the content should be used, conditions can be applied to text, and filtering used to selectively remove text from outputs for a particular market, audience or publication.

In the following example, the measurements have been provided in two different units, so that the text can be understood by readers from the United States (and other countries using Imperial measurement units) and readers who use the metric system. The text has not been written in the context of the audience using the metric system.

> The extender adds approximately 8 inches (200 mm) of length and it can be used for either the driver or front passenger seating position.

Although it might appear that the better outcome might be one where the source contains both measurements, that is not necessarily the case. DITA and other semantic XML approaches incorporate conditional filters which can be used to exclude the non-

preferred unit from the output for a particular market. However, even in a country using the metric system there will be drivers who better understand the Imperial system.

Filtering can also enhance readability by removing clutter. In the example below, the comprehensibility of the text is impeded by the condition that the sentence only relates to cars fitted with an immobiliser. Using conditional filtering techniques, this sentence could be excluded from manuals for car models without immobilisers, and the parenthetical clause deleted.

> The security indicator light will continue to flash once every three seconds indicating that the system is in the valet mode (only vehicle with an immobiliser).

Removing context in graphics

Illustrations, photographs, images and other graphic devices within documents are themselves innately modular, in that they are stored as separate files, and can be used in different contexts in different documents. However, the content of graphics can bind them to context. For example, a photograph with a figure title superimposed on the image can only be re-used if the figure title is relevant in the different re-use context. A photograph with English language call-outs can only be used in an English language document.

Care taken in the creation of images can maximise their re-use potential. The omission of branding from graphics will permit greater re-use. A photo identifying the radiator grille of a car is not significantly enhanced by the inclusion of the car maker's bonnet badge, but including the badge in the photo will render it unsuitable for illustrating any document other than one concerning that car brand.

Removing sequence

Use of terms that denote sequence is another form of context that must be removed for a text to be context-agnostic. Adjectives such as "previous", "next", "earlier", "aforementioned" and "later", as in "more information on the product limitations are discussed in later chapters", should therefore be avoided. Including such restrictive references introduces the risk that the blocks of content won't be in the sequence in all output publications, and that the adjective becomes incorrect and misleading. (It is also possible that the referenced blocks might not be included in all output publications.)

While adjectives such as "following" or "preceding" do apply a context, that fact will not usually present a problem if the referred element is within the same information chunk. (A *chunk* is a block of text describing one idea or step.) For example, if the lead-in sentence will only be re-used in conjunction with the larger block in which it is structured, then the context is confined to *intra-chunk*. Context-agnostic writing only needs to aim to minimise *extra-chunk* context, as this type of context is limiting to re-use.

An example of *intra-chunk* context is:

> Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

In a semantic mark-up language such as DITA, the mark-up itself can be used to bind the lead-in sentence to its antecedents, as in the example:

```
<p>Operation is subject to the following two conditions:
 <ol>
  <li>This device may not cause harmful interference, and</li>
  <li>this device must accept any interference received, including
  interference that may cause undesired operation.</li>
 </ol>
</p>
```

The paragraph element (<p>) contains both the word "following" and the items that follow. Whenever the paragraph is re-used, the items will be included in the re-use.

Manually-applied numbers are also problematic, as they lock in a sequence. If enumerations such as chapter, figure and paragraph numbers are required in the output document, they should be automatically generated through the publishing process. (The publishing process converts the topics from their authoring format to a deliverable document in a reading format such as paper or Web.)

While it may initially appear that having no numbered identifiers in the authoring format would make cross-referencing (such as "listed in Table 3—Codes") impossible, in fact cross-referencing is a trivial authoring task. DITA incorporates a comprehensive cross-referencing mechanism using metadata as the linking key. This allows references to numbered objects in the output to also be automatically derived during the publishing process.

Cross references

Although cross-references may be technically easy for a writer to devise, cross-references are heavily burdened with context.

Cross-references to other parts of a publication are context-specific, because to make sense they require a pre-condition that the referenced target (in the example below, the "Activating . . ." procedure) is included in the output publication.

> To exit valet mode, change the setting of your vehicle's alarm system for activation mode. (Refer to "Activating and deactivating the alarm system" in this section.)

Cross-references are, however, often critical to the navigation logic and the understandability of the text. Rather than remove this context completely, the best approach is to move the context from the topic to the *document map*.

The document map (in DITA it is known as the *ditamap*) is a *manifest* listing the topic modules that are to be included in a deliverable document, and the hierarchy and sequence in which they will appear. A ditamap is therefore specific to a publication, while topics may be re-used in many publications.

Specifically to allow the shifting of link context from context-agnostic topic to context-specific map, DITA includes a *relationship table*, defined in the ditamap. The relationship table defines the linking relationship between topics in the publication. Because the ditamap defines the collection of topics to be delivered as a publication, this is a more logical place for links between topics in the collection to be defined. When the ditamap is processed to create the deliverable document, the relationships in the relationship table are translated into cross-references that are added at the end of the topic in the output.

There is some evidence that, for Web content, cross-reference links placed at the bottom of the page (i.e. outside the paragraph text) may be more usable than inline hyperlinks (McGovern, 2009)—an added benefit derived from the use of relationship tables.

Externalising other content

XML-based modular writing systems offer other methods to *externalise* context, or move the context from topics to the map manifest. One such method is the *keyref* feature offered in DITA.

The concept of keyref is that references to other resources are made with one level of abstraction; the link in the topic refers to a key, and a matching key in the ditamap refers to the resource. This concept is known as *indirection*.

For example, rather than a cross-reference in a paragraph directly referring to another topic, such as in <xref href="wrx_specs.dita">, the cross-reference would refer to a key, as in <xref keyref="model_specs">. When a direct reference (in the example, to the Subaru WRX model specifications in wrx_specs.dita) is used, the paragraph can only be re-used in a WRX publication. However, when an indirect reference is used, the actual target is defined in the ditamap file which specifies the publication. The code in the WRX ditamap might be <topicref keys="model_specs" href="wrx_specs.dita">, while in the ditamap used for the Saab 9_2, the code might be <topicref keys="model_specs" href="9_2_specs.dita">. Using this indirection technique, the cross-reference in the one topic can refer to entirely different target topics depending on the ditamap in which the topic is used. Indirection can be used for any addressable document component: links, cross-references, glossary definitions, images, and content snippets.

Indirection effectively moves context from the content topic to the document specification (map) level.

Conclusion

The change in the technical communication field from document-centric writing to topic-based writing, made practical by the development of XML-based documentation architectures such as DITA, will necessitate a change in writing approach to separate content, form and context. The modular DITA architecture in particular provides for the creation of independent content *topics* that are assembled into deliverable publications through document *maps*. While semantic markup allows content to be separated from form, features in the document map allow some types of context to be moved from the topic level to the map level.

Adopting context-agnostic writing techniques to topic-based modular documentation will result in greater content re-use opportunities, leading to improved efficiency through the technical documentation life cycle. Those writing techniques include:

- removing or minimising context phrasing;
- avoiding supra-organisational specific terms;
- using filtering to selectively remove conditional content;
- removing branding and other context from graphics;
- removing sequence context from topics, and automatically applying sequence in the publishing process; and

- externalising cross-references through relationship tables, variables and indirection.

Context-agnostic writing does not mean the removal of context, but the separation of context from content and form. While it is not known whether context-agnostic writing will affect the quality of communication by leading to an inferior, superior or equivalent experience for the reader, the concept of re-introducing context customised to the requirements of the individual reader, at the point of document delivery, holds much promise.

The prime benefit of writing modular content suitable for publication in different forms for different purposes in different contexts is efficiency—being able to create more in less time. This benefit is not only applicable to the field of technical communication, but also in other areas of written communication practice where the drive for greater efficiency suggests a re-evaluation of writing processes. By adopting context-agnostic writing techniques for topic-based modular documentation, technical writers can improve content re-use and achieve greater efficiency through the technical documentation life cycle without significantly compromising the quality of the reading experience.

References

Ament, K. (2003). *Single sourcing: Building modular documentation*. New York: William Andrew.

Hackos, J. (1994). *Managing your documentation projects*. Wiley Technical Communication Library (1st ed.). New York: John Wiley.

Hackos, J., & Hedlund, T. (2001). *Making a business case for single sourcing.* Denver, Colorado: Center for Information-Development Management. Retrieved May 15, 2010, from http://www.comtech-serv.com/pdfs/Business_Case_for_Single_Sourcing.pdf

Houlihan, D. (2008). *The technical communicator's transformation: Publishing on time and on quality.* Boston: Aberdeen Group.

Houlihan, D. (2010). *Benchmarking content reuse in technical communication.* Boston: Aberdeen Group.

McDaniel, E. (2005). Consider the source: Structured authoring for XML-based documentation. In *UNC College and University System Exchange (CAUSE) Conference, Raleigh, North Carolina, University of North Carolina*. Retrieved February 23, 2010, from http://www.unc.edu/cause05/presentations/mcdaniel/mcdaniel.pdf

McGovern, G. (2009, November 2). Why web links are calls to action. *New Thinking*. Retrieved February 23, 2010, from http://www.gerrymcgovern.com/nt/2009/nt-2009-11-02-Web-links-action.htm

O'Keefe, S. (2006, April 12). Wednesday at WritersUA: Structured authoring: taking the plunge. *Scriptorium*. Retrieved May 15, 2010, from http://www.scriptorium.com/blog/2006/04/wednesday-at-writersua-structured-authoring-taking-the-plunge.html

O'Neill, J. (2002). A global style guide: Working together around the world. In *STC 2002 Conference Proceedings*. Retrieved August 26, 2009, from http://www.stc.org/confproceed/2002/PDFs/STC49-00024.pdf

Petelin, R., & Durham, M. (1992). *The professional writing guide*. Warriewood, NSW: Business and Professional Publishing.

Robidoux, C. (2008). Rhetorically structured content: developing a collaborative single-sourcing curriculum. *Technical Communication Quarterly, 17*(1), 110-135.

Rockley, A. (2001). The impact of single sourcing and technology. *Technical Communication: Journal of the Society for Technical Communication, 48*(2), 189-199.

Sapienza, F. (2004). Usability, structured content, and single sourcing with XML. *Technical Communication: Journal of the Society for Technical Communication, 51*(3), 399-408.

Williams, J. D. (2003). The implications of single sourcing for technical communicators. *Technical Communication: Journal of the Society for Technical Communication, 50*(3), 321-327.

WritePoint. (2010). *DITA challenges survey results (November 2009)*. Commercial. Retrieved January 18, 2010, from http://www.writepoint.com/site/downloads/WritePoint_DITASurveyChallenges_Results.pdf